Discussion DRAFT November 16, 2003

Shared Data Model for Describing Plans and Regulations as Inputs to Simulation Models of Urban Development

Lewis D. Hopkins, Nikhil Kaza, and Varkki G. Pallathucheril
University of Illinois at Urbana-Champaign

Prepared for 2003 North American Meetings of the RSAI,
November 23, 2003

Abstract. A shared planning data model for describing inputs and interpreting outputs of simulation models of urban development will yield three benefits. First, it will enable users to specify scenarios involving more complex, dynamic, and endogenously triggered plans, policies, and regulations. Second, it will enable users to use more than one model in order to consider the implications of potentially different results from partially substitutable models of the same phenomena. Third, it will enable users to link models of different phenomena such as transportation, land use, environmental effects, demographic change, and economic change. This paper will report an early version of a "planning data model" with particular focus on its fit with the UrbanSim and LEAM models of urban development. The intent is to invite response so that a more general and effective common data model can emerge.

Urban development models are used to trace out the implications of particular ideas about development processes, initial conditions, and possible trends. Plans and regulations are pertinent to these models in at least two ways. First, development processes being modelled respond in part to regulations and plans of the various municipalities and agencies in the region. Second, a frequent application of these models is to predict the effects of a proposed plan or regulation and compare these to the effects of other plans and regulations.

A data model for plans must consider both the logic of plans and the content of such plans. Thus, a data model for plans and regulations inherently includes the representation of the processes and phenomena of urban development. Urban growth boundary regulations, for example, could not be expressed without also expressing the supply of land resulting from changes in developed land at specific locations. The Planning Data Model (PDM) presented here is, therefore, intended to provide a shared data model encompassing all variables in urban development models, not just plans and regulations. The PDM can thus provide three benefits. First, it will enable users to specify scenarios involving more complex, dynamic, and endogenously triggered plans, policies, and regulations. Second, it will enable users to use more than one model in order to consider the implications of potentially different results from partially substitutable models of the same phenomena. Third, it will enable users to link models of different phenomena such as transportation, land use, environmental effects, demographic change, and economic change.

The PDM will enable plans to be coded as inputs to urban development models and other impact assessment models. Models of urban development processes currently used to assess transportation, housing, land use, land cover, air pollution, and other environmental consequences can test only initial policy conditions, not dynamic interactions of plans and regulations with the processes of urban development over time. A common data model for representing such complex plans and regulations will enhance the capabilities of currently available urban development models. For example, if an urban growth boundary must meet legislative requirements for available land, the growth boundary will change over time as the urban development process occurs. If a capital improvements program is based on triggers of minimum available capacity, then capacity constraints in the model must change as development occurs. If land parcels are rezoned in response to petitioners as development spreads and land is annexed, then zoning should be represented dynamically, not as initially fixed constraints.

The PDM will also enable "multiple modelling" as an approach to knowledge discovery and decision making. A logical data model for planning must be able to represent states of the world with respect to urban development in a way that makes sense for plans, regulations, and forecasts. It will also, therefore, be sufficient to encode the entities and processes in simulation models of urban development and make multiple modelling possible. There are two ideas underlying multiple models: model combination and model triangulation (Hopkins 2003). To combine models, common definitions of inputs and outputs, and in some cases common endogenous variables or entities, must be achieved. The central idea of model triangulation is based on the procedure from surveying of determining the location of a point from the known locations of two other points. We learn more from trying to address a question in distinctly different ways. To consider what the different models mean—whether they replicate and confirm, contradict and yield insight, complement and reinforce, or converge and enable action—requires some way to relate models to each other. A common data model is essential to enable transformations of the inputs and outputs of different models so that models can be combined or so that contrasts and comparisons are meaningful.

This paper is organized as follows. First, a conceptual planning data model is described encompassing urban development processes, plans, and regulations. Second, this data model is compared to the data structures of UrbanSim and LEAM, two quite different urban development models, to demonstrate its generality. Third, examples of complex and endogenous plans and regulations are expressed in terms of the PDM to illustrate the potential to incorporate these into simulations. Fourth, a multiple modelling example illustrates how the same question might be asked of UrbanSim and LEAM. This is a very early version of the PDM and is presented to invite response toward its development. The example applications are intended only to illustrate the ideas, not to test the data model.

A Data Model for Urban Development Planning

Plans describe actions that might be taken in certain circumstances and thus change the state of the world. The state of the world is dynamic, not static and includes legal rights, human capabilities, and social norms, not just activities and physical entities. Possible actions include investments and regulations. Plans must be represented so that they can be applied in particular situations as agendas, visions, policies, designs, and strategies. This overall framework is diagrammed in Figure 1. Figure 2 is a diagram of entities for urban development modelling proposed by Waddell and Ulfarsson (forthcoming), which we have annotated with the labels for these classes of entities in the current version of the PDM to show its scope for urban modelling as well as for representation of plans and regulations. These entities and relationships are elaborated here, focusing first on the world and then on plans and regulations.

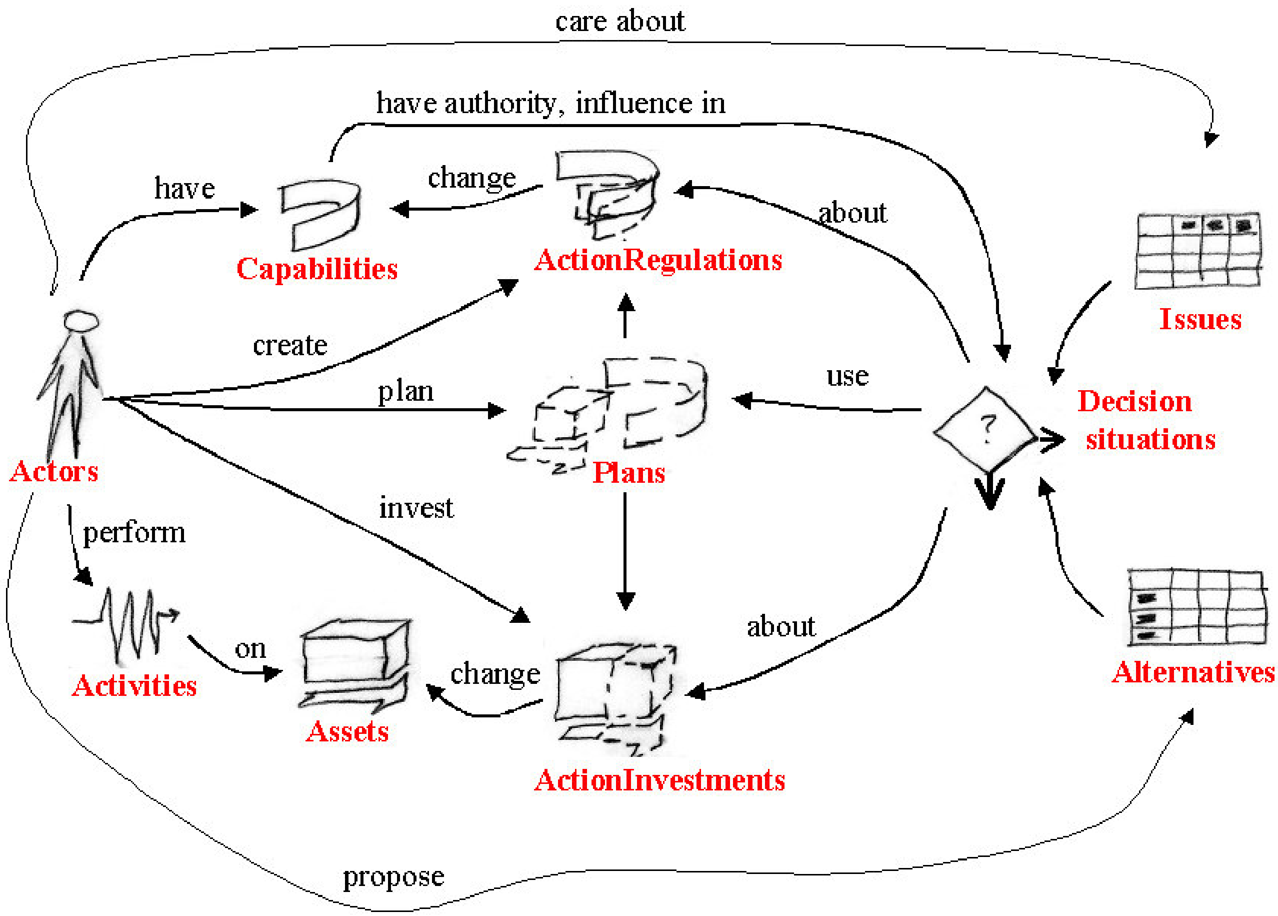

Figure 1 Entities for Urban Development

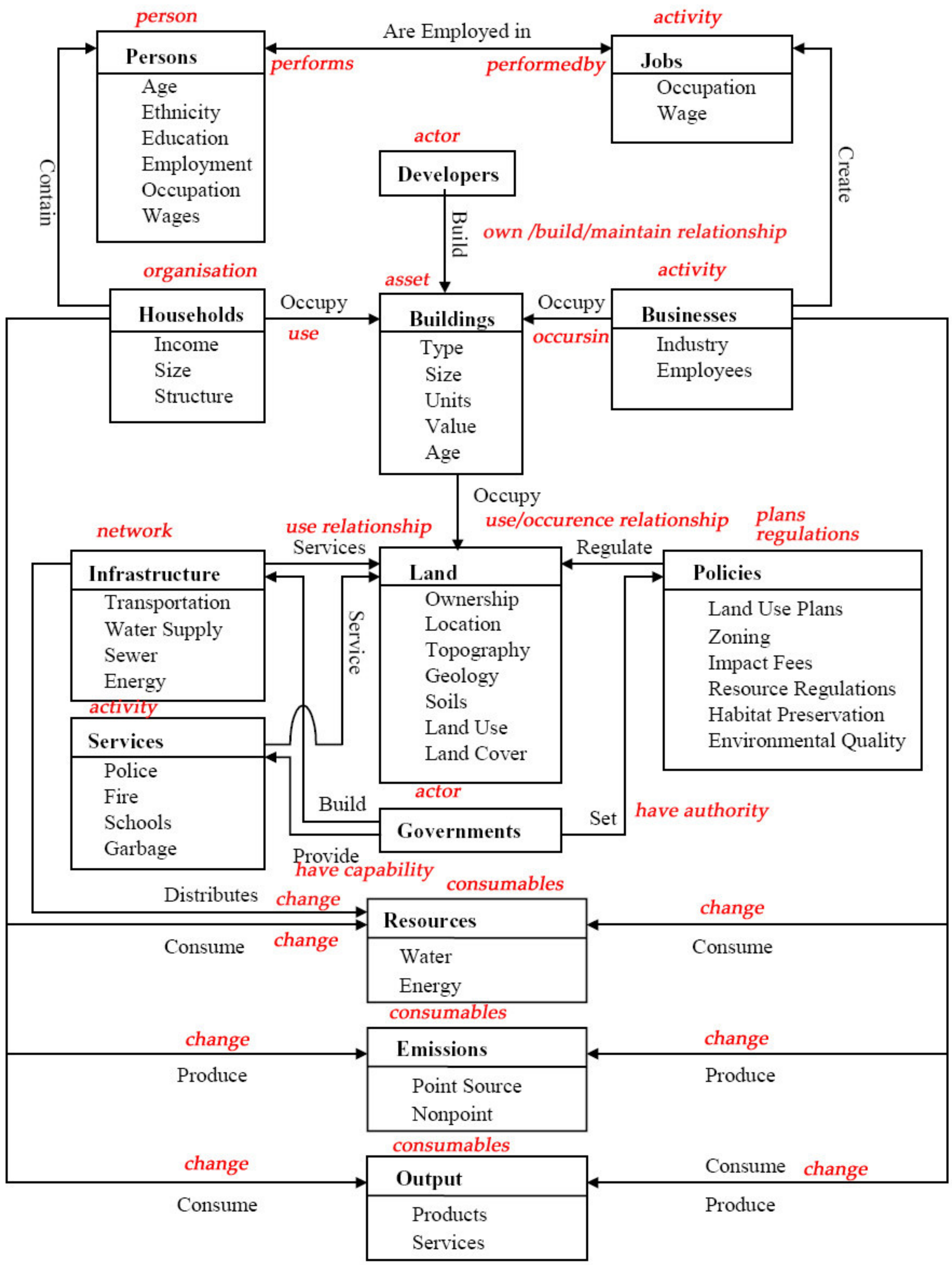

Figure 2 Annotation of urban development model entities
as described in Waddell & Ulfarsson (forthcoming)

Actors, Activities, Assets, Actions and Relationships

Actors are Persons, Organizations, or Populations of Persons or Organizations as diagrammed in Figure A.1 in the Appendix. A group of Persons organized in Roles, Responsibilities, and Decision Rules is an Organization. So, for examples, households, firms (in the economic sense), neighbourhood groups, government agencies and city councils are Organizations. Populations are collections of Actors without organizational structure, such as the Population of Persons in a census tract or the Population of firms in a municipality. Actors have Roles and many of the Capabilities of Actors are associated with Roles rather than directly with Actors. For example, the Authority of a mayor goes with the Role, not the Person. Also Roles can exist without an Actor associated with them, so that the authority of a Mayor is defined regardless of the Person holding the office, but the influence a particular mayor may have depends both on the Role and the Person. Similarly an Actor can have multiple roles the combination of which will determine the set of Capabilities the Actor possesses.

Activities occur on Assets and are performed by Actors as diagrammed in Figure A.2. Activities are aggregates of behaviour occurring on Assets performed typically by Populations of Actors. Traffic on a street network (commuting), shopping by a Person, and retail services in a building are Activities. Activities are different from Actions in that Activities describe aggregates of behaviours that are not fundamental changes to the system of Assets and Capabilities and for which Decisions to act are not explicit. Activities are also constrained by Capabilities of Actors but it might not always be possible to identify a one to one relationship between Activities and Actors. An Activity may have effects on Assets, notably depreciation. Activities are also subject to capacity constraints and congestion relative to Assets

Assets can be Facilities, Equipment, Consumables or Intangible as diagrammed in Figure A.3. Facilities are Physical objects such as building Structures or Networks, such as streets. They can also be Virtual Networks such as microwave networks or Designated Areas such as land zoned for development or protected habitats. Assets are related to other Assets. For example, Equipment may be assigned to a particular Facility. Land or water in a river could be defined as an Asset from which resources are used. Buildings could be located on a site or a dam on a river at a location at a time or for a period of time. Actors in their Roles can own, lease, hold government jurisdiction over, have maintenance responsibility for, or have other use rights in Assets.

Actions, not to be confused with Activities, change Assets themselves or their relationships to Activities or Actors as diagrammed in Figure A.4. Actions are central to the planning domain and include decisions and realized actions. Decisions are commitments to Actions that have not yet been realized. Thus a Decision by a city council to invest in a road project is distinct from the realization of that project on the ground. Decisions and realized actions include Regulations, Investments and Transactions. Actions can also change Capabilities of Actors and include changing Rights and Responsibilities. It is useful to distinguish between realized Actions and Decisions as commitment to Actions, because responses to Actions by other Actors may be based on Decisions or expected actions before an Action is realized. Actions have consequences either realized or expected, which are generally distributed over space as well as time. The consequence themselves are represented as states of the world.

Plans and Regulations

The data model also represents plans and regulations. Plans are elaborated as working through aspects: Agenda, Policy, Vision, Design, and Strategy. These ideas are fully developed in (Hopkins 2001) and diagrammed in Figure A.5. Each of these defines a particular kind of relationship among Actions and among Actions and Consequences in a Plan. An Agenda is a list unrealized. A policy is a set of conditions in which an Action will be taken. A Vision is a state of the world expressed as an aspiration. A Design is an interdependent set of Actions with respect to Consequences. A Strategy is a set of contingent Actions and uncertain Consequences. Some elements of plans may be represented in more than one way. For example, particular Investments in roads might be represented as an Agenda in a capital improvements program and as a Design for a Network in a transportation plan. Regulations are similar in form to policies, but regulations are enforceable as defining or redefining rights. A more complete explanation of the plan components of the data model is presented in Hopkins, Kaza, and Pallathucheril (2003). These ideas are illustrated further in the following explanations of the scope of the PDM.

Modelling with Plans and Regulations

It is important to acknowledge behaviour and strategies of other players when evaluating and formulating ones own strategies. Other players may plan and regulate. The other players can be different agencies of government including adjacent municipalities, special districts, and other levels of government. To simulate the impact of a plan by Portland Metro, a model should take into account the plans, regulations, and behaviours of municipalities and special districts within the region, as well the rules by which Metro itself plans. Major developers and major employers may also follow plans and policies that should be modelled. If the rest of the world is planning, then urban development models should include plans. If many different actors are modifying regulations in response to emerging situations, then urban development models should include regulations.

Most urban simulation models presume that an existing state of the world and a policy regime are given *a priori*. A scenario is constructed as combinations of different plan elements, such as transportation investments, or regulations, such as zoning or urban growth boundaries. The simulation models then predict only the effects of certain choices of actions from this starting point, which remain fixed. The processes of urban development include error control feedback mechanisms, however, which plans fixed *a priori* cannot represent. Information gained by actors over time will result in changes in preferences and behaviours and urban simulation models should be able to represent these phenomena in order to provide useful predictions.

Consider the case of a city forecasting its need for infrastructure based on population estimates. The forecast is more useful if the city takes into account a proposal by a large corporation to build a manufacturing plant or the Department of Transportation to build a new expressway. These examples underscore the need to take into account plans of crucial players in the development process. Building into the model the knowledge or perceptions of proposals of other actors is valuable in formulating appropriate actions.

In a tiered system of governance, one actor is required by another to perform certain actions. It is thus necessary to be able to incorporate these requirements into a model so that it is possible to construct a scenario in which these requirements can be modified. A regulation put in place by state government about redrawing an urban growth boundary has an effect on growth patterns and alternatives available to the city. In Oregon, an urban growth boundary must be re-evaluated every 5 years to accommodate the forecasted growth for 20 years (Knaap and Hopkins 2001). To represent this regulation of agencies responsible for UGBs, a model should be able to represent these regulations using defined data classes. We do not in this paper distinguish carefully between regulations and policies structurally but the instruments have entirely different effect, application and enforcement mechanisms (Hopkins 2001; Hopkins, Kaza, and Pallathucheril 2003)

Plans could be incorporated into urban simulation modeling in at least three ways. First, a plan or regulation could be fixed *a priori.* For example, certain locations could be identified before the model is run as places that will be more likely to develop in particular ways because the plan says they should. LEAM and UrbanSim accomplish such fixed plans through a "policy layer," which for example identifies exogenously the plan designation of each cell or whether a cell is inside or outside the urban growth boundary. This expectation based on the plan could not, however, change as development occurred even if in the real world such a plan would be likely to change in response to development. Second, a plan could be dynamic and be adjusted interactively by a model user during a run of the simulation model. In this case, the user would monitor indicators or development patterns at each recursive time step and decide whether and how to adjust the plan to imitate real world behaviours of plan evolution. Finally, a plan could be endogenous to the model run, responding to values of variables computed while the model is running. The process of adjusting an urban growth boundary regulation, which is elaborated in later sections below, illustrates this case.

The policy aspect of plans serves as an illustration of the PDM. A Policy is an if-then statement, which is applied repeatedly given a situation. A unified modelling language elaboration of a policy data model is shown in Figure 3. The given situation (the if clause) could be about attributes of any entity in the data model including Actions by Actors. The policy to be applied (the then clause) also depends on Capabilities of the Actors following the Policy. The Actor to whom the Policy applies, who may be different from the Actor who created the Policy, takes the then statement. For example, the structure of the UGB case can be represented in the following pseudocode:

```
Policy about Urban Growth Boundary
Owner = State Government
Time of adoption = mm/dd/yy
Temporal span = xx years

If
   area is designated as urban growth boundary ( attributes of state of
world)
   And development is proposed outside UGB     (action)
```

```
 then
     deny permit;                                  (action)
    action enforcer = city & county governments;    (actor)

If
   area is designated as urban growth boundary
 then
    action = revise UGB;
    criteria = every 10 years asses forecast of land requirements and include
10% area extra.
    action enforcer = Metro regional service district;
```

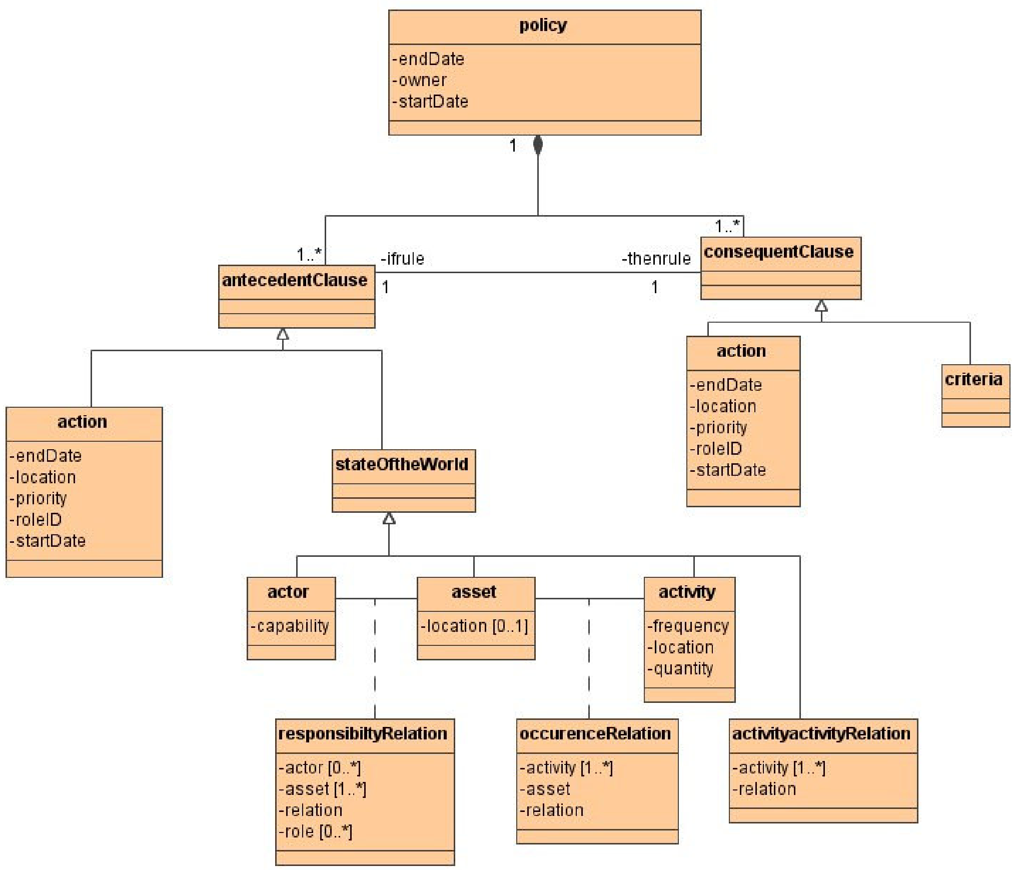

Figure 3 A class diagram for policy aspect of plans or regulations

It should be noted that there is no restriction on the policy variables. A policy need not be geographical in nature. A policy can be about adoption of a regulatory mechanism, a tax structure or Investment decision. A policy could be about improving the quality of schools, which will act as an attractor for residential development. Such a non-spatial policy has a significant land use effect but is rarely captured in current land use models. Naturally the geographical extent of applicability of a policy is limited by the jurisdiction of an actor following the policy or the scope of the model. Policies however can be very specific to spatial locations, for example, insurance requirements in flood zones along a particular water body.

It is important to be careful that endogenous plans and regulations do not confound interactions among sub-models of the urban simulation model. This separation should be relatively straightforward in object-oriented models where sub-models remain largely independent of the data and other models and interact iteratively and recursively by maintaining an object store or a data store and checking regularly for any changes in the data objects. Thus incorporating endogenous policies into a model should not translate into a tighter coupling of sub models or interaction mechanisms.

The data model should and does allow for changes not only in the choice for single actions, but also choices about structures of action. A choice of policy (if-then rules) is different from a choice of strategy (a set of contingent policies), which is different from a choice of regulation (a definition or redefinition of enforceable authority available to others). Each of these sets a structure of action within which individual actions are taken.

Data Model Interpretation for UrbanSim and LEAM

The PDM is intended to encompass the scope of data entities used in a wide range of urban development models (See for example, United States Environmental Protection Agency 2000). Here we focus on UrbanSim (Waddell 2002) and LEAM (Deal and George 2001) because they are quite different and thus illustrative of the desirable scope of a shared data model.

UrbanSim

UrbanSim is an urban simulation model that includes land use, transportation, and environmental impacts. It is object oriented in conceptual design with explicit recognition of actors, development, and economic concepts of pricing. Its system architecture, as diagrammed in Figure 4, relies on a data store of objects, a translation layer, and a set of models that draw on these data objects in a process of recursive simulation. Figure 4 is also annotated to show that the object classes in the PDM encompass the classes in UrbanSim. Table 1 also shows this correspondence of object classes. The next step will be to test this correspondence by specifying a run of UrbanSim in which the initial dataset and the scenario dataset are expressed in terms of the PDM. In particular, it may be possible to express more complex policies than can currently be expressed as input to UrbanSim, such as capacity expansion strategies contingent on future states of the world or on future forecasts of states of the world.

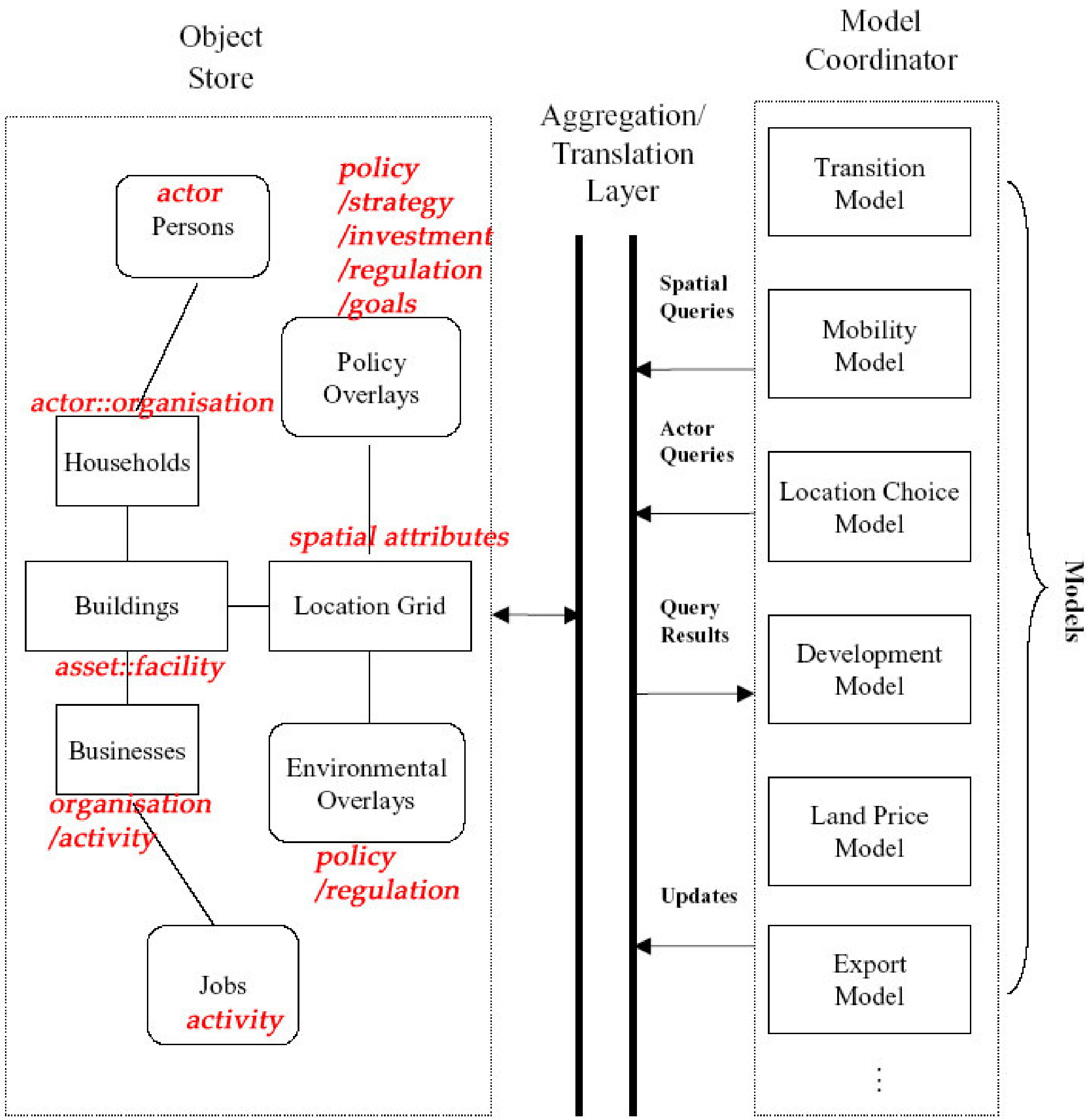

Figure 4 UrbanSim data structure from (Noth, Borning, and Waddell 2003) annotated with classes from planning data model

UrbanSim Object		PDM equivalent class
Persons		Actor:: Person
Households		Actor::Oganisation
Buildings		Asset::Facility::Structure
Businesses	Individual or organized collection	Actor::Oganisation
	As use or economic activity	Activity
Jobs		Activity Collection
Housing preferences		Capability::Preference of an Actor Collection
Housing stock		Activity Collection
Business Sector		Activity Collection
Commercial Space	Zoning Category	Activity constrained by Capability::Authority of Actor by Action::Regulation of Actor::CityGovernment
	Existing land use	Activity on Asset::DesignatedArea
County area		Capability::Jurisdiction of Actor::Organization::County Government
Urban Growth Boundary		Asset::DesignatedArea created by Action:: Regulation under the Capability:: Authority of Actor::CityGovernment in Capability::Jurisdiction
Transport Link		Asset:: Facility::Network::Link
Trip		Activity
Vacancy rate		Indicator (function of Activity on Asset in DesignatedArea)
Access to the nearest airport		Indicator (relationship of Actor or Activity to Activity or Asset::Facility::Airport)

Table 1 Comparison of UrbanSim Objects and Planning Data Model classes

LEAM (Landuse Evolution and Impact Assessment Model)

LEAM is a hybrid approach to modelling urban development, and combines regional drivers of land-use change (the economy, for instance) with cell-based drivers (access to jobs, for instance). It combines use of STELLA for constructing the local rules that drive cellular change, and the Spatial Modeling Environment (SME), developed at the University of Maryland, for spatializing the cellular models. STELLA is a graphically based dynamic simulation software based on Jay Forrester's systems dynamics language that uses icons and symbols to communicate a model’s structure (Forrester 1961). Icons include reservoirs representing stocks of resources and "pipes" and "valves" representing flows and controls between those reservoirs, each with an associated user defined equation (Hannon 1994). SME spatializes the single-cell STELLA models, applying them to a geographic area (represented as a matrix of cells) and simulating the changes that take place to the state of each cell over multiple time steps. SME automatically converts the STELLA models into computer code that can be run on multiple processors (and multiple computers) in parallel.

Figure 5 describes the LEAM approach to simulating land-use transformation dynamics. It begins with model drivers, which are those forces, typically human, that contribute to urban land use transformation decisions. The model drivers individually produce land-use transformation probabilities for each cell that are then combined across drivers. Change from existing to new land uses is then predicted based on demand control totals, aggregate transformation probability, and simulated 'luck of the draw.' The resulting land-use pattern is then analyzed for environmental, social, and economic impacts. Impact models, which produce impact indices that can be compared to sustainability benchmarks, may feed back into the model drivers or simply report indicators of the state of the system at a time interval.

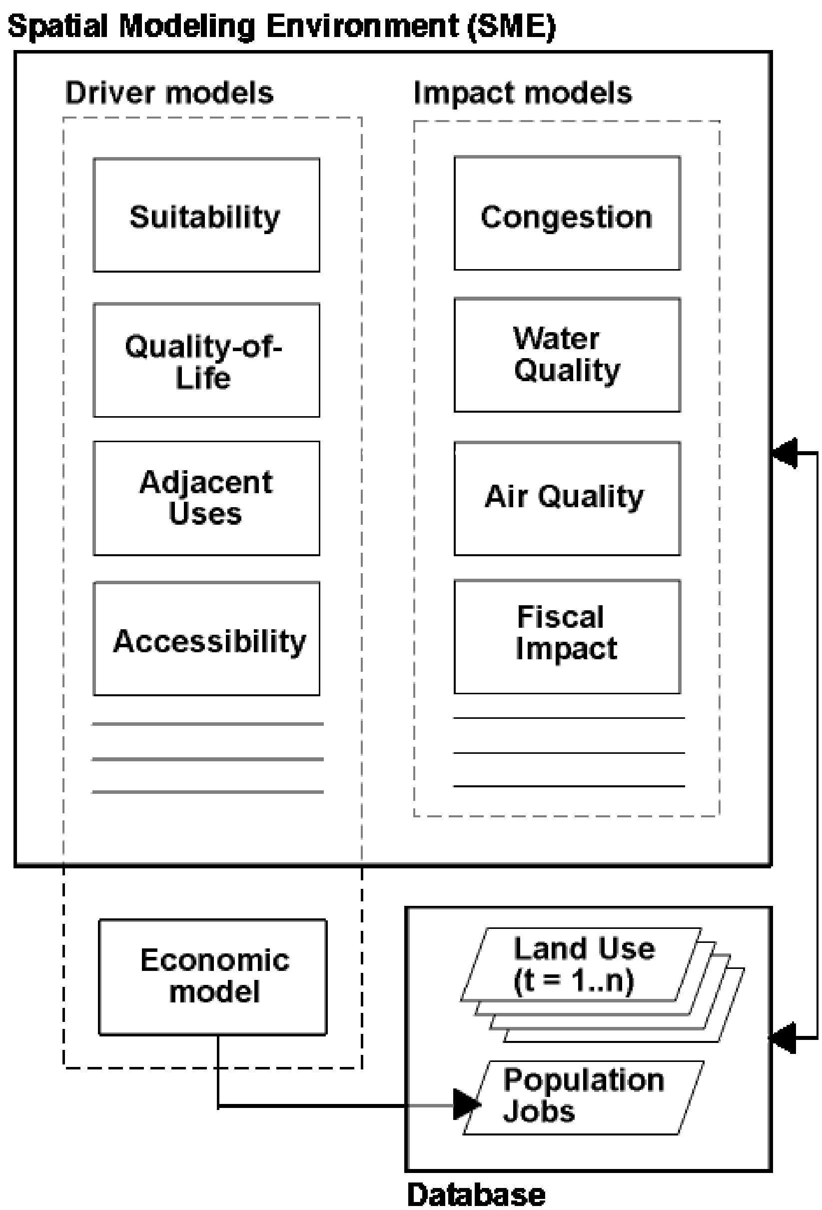

Figure 5 LEAM interaction Diagram

Table 2 compares the key variables that go into LEAM drivers with the PDM and shows that object classes in the PDM can represent these variables. For LEAM variables, objects must carry with them information about how they affect land-use transformation probabilities. Alternatively, at a higher level of abstraction, an object class could be a description of the driver system and how it interacts with other drivers. This suggests a limitation of the current PDM with respect to simulation models: The PDM lacks object classes for model management.

LEAM variables (partial list)	**PDM equivalent class**
Jobs	Activity Collection
Existing land use	Activity::Landuse on Asset::DesignatedArea
Household size	Attribute of Actor::Organisation::Household
Vacancy rate	Indicator (function of Activity on Asset in DesignatedArea)
Transport Link	Asset:: Facility::Network::Link
Traffic congestion coefficient	Indicator (function of Activity on Asset)
Infrastructure service area	Asset::DesignatedArea derived from Spatial relation (located in the range) of another Asset::Network::Infrastructure
Access to infrastructure	Indicator (Actor or Activity Relationship to Asset)
Slopes	Attribute of Asset::DesignatedArea
Floodplains	Asset::DesignatedArea
Urban Growth Boundary	Asset::DesignatedArea created by Action:: Regulation under the Capability:: Authority of Actor::CityGovernment withn Capability::Jurisdiction

Table 2 Comparison of some LEAM variables and Planning Data Model classes

Learning from Simulations with More than One Simulation Model

Decision makers value a second or third model or forecast either because it yields a second opinion or because it adds aspects to the first model so that the combination of models is more complete. The most useful second opinion is one that is clearly independently derived, but meaningful in contrast and comparison to the first. Whether focusing on models directly, or on the forecasts these models imply, it is tremendously valuable to be able to get a second opinion. Often, a third and fourth opinion will be offered and may well be valuable. To make use of multiple models, there must be a way of placing them in a common framework, even if that framework serves to show that the models are indeed not about exactly the same thing. Recognizing these differences is often particularly useful in gaining insight (Hopkins 2003). The proposed research will yield a common data model that will make possible comparison and contrast of simulation model inputs, outputs, and in some cases intermediate entities in simulations model processes and calculations.

In addition to the idea of partial substitution of one model for another, models are often used in combination, frequently in sequence, recursion, or iteration. Westervelt (2001) describes a modelling environment that supports linking models developed by many different teams working separately and in different disciplines. The mechanisms for linking models include spatial, functional (process), and temporal frames. The premise is that there are many, disparate models available to address aspects of a watershed system, for example. We can leverage these modelling assets by creating computing environments in which to use them. A first level is to provide a common input and output frame (a data model) that can encompass different models. Even more powerful will be a frame in which different models can interact as processes, not merely as inputs and outputs. Both LEAM and UrbanSim are structured to combine multiple models through a common environment, SME for LEAM and Model Coordinator for UrbanSim.

Figures 6 and 7 show LEAM and UrbanSim data types in similar form suggesting that input data and output data could be translated through or from the Planning Data Model so that comparable simulation runs for the same scenario could be run on both simulation models. Consider an urban growth boundary regulation. Different actors are involved in making, revising and enforcing the regulation. The idea is to represent the UGB regulation in the PDM so that both LEAM and UrbanSim can access the same UGB characteristics, as shown in the following pseudocode:

UGB REGULATION

Antecedent Clause

```
     Development(Investment)is proposed(action)by
developer(actor)
And(logical operator)
   It is located at (X,Y)(spatial attribute)
And (logical operator)
     (X,Y) is Outside (spatial relationship GML operators) of
UGB (designated Area  GML data)defined for a City (actor)
```

/*This is an asset-asset spatial relationship between the asset resulting from the investment and the designated area asset.*/

Consequent clause

```
     City Government (actor) will deny permission (Action)
under authority(capability) granted from enabling legislation
(regulation)of state government(actor) Z (pointer to regulation)
```

This pseudocode relies on the existing shared data model for geographic operations expressed in the Geography Markup Language (GML) (Lake 2000) for spatial operations, rather than presuming to create geographic concepts in the Planning Data Model. The PDM should eventually be XML compliant in order to be used on the web, so it makes sense to use XML compliant elements such as GML that already exist. UrbanSim, when accessing this policy, would then take the GML compliant data and spatial operator to determine in its gridded dataset whether the cell is inside or outside the UGB of a given city. The entities in UrbanSim necessary to do these operations and the PDM equivalents are shown in Figure 7. LEAM on the other hand will simply take the GML data and Grid it on the lines of the other data sets and use it as one more data set to consider when deriving the probability of development and setting the probability of development to 0 with a large weight outside the UGB. The LEAM data items and PDM equivalents are shown in Figure 6. Even though fundamentally different data structures are used in the two models, both can access a regulation specified in the Planning Data model.

Gridded Data sets	
Landuse Type	*asset::designatedArea*
Slope	*attribute of asset::natural resource*
school district	*spatial relationship of capability::jurisdiction of actor::organisation::School District*
Proximity	*spatial /functional relationship*
Infrastructure availabilty	*determined by functional relationship with asset::network*
Roadshed	*asset::designatedArea determined from a spatial operation on asset::network*
Watershed	*asset::designatedArea*
UGB	*determined by spatial relationship(located within) with jursidiction*
City	*determined by spatial relationship(located within) with jursidiction*

Non-Gridded Data sets	
Road Network	*asset::Network*
habitat fragmentation	*asset::Natural Resource*
Regional Transportation	*asset::Network*

Population	*actor::population*
Jobs	*Activity*

Figure 6 LEAM input data entities and representations in Planning Data Model

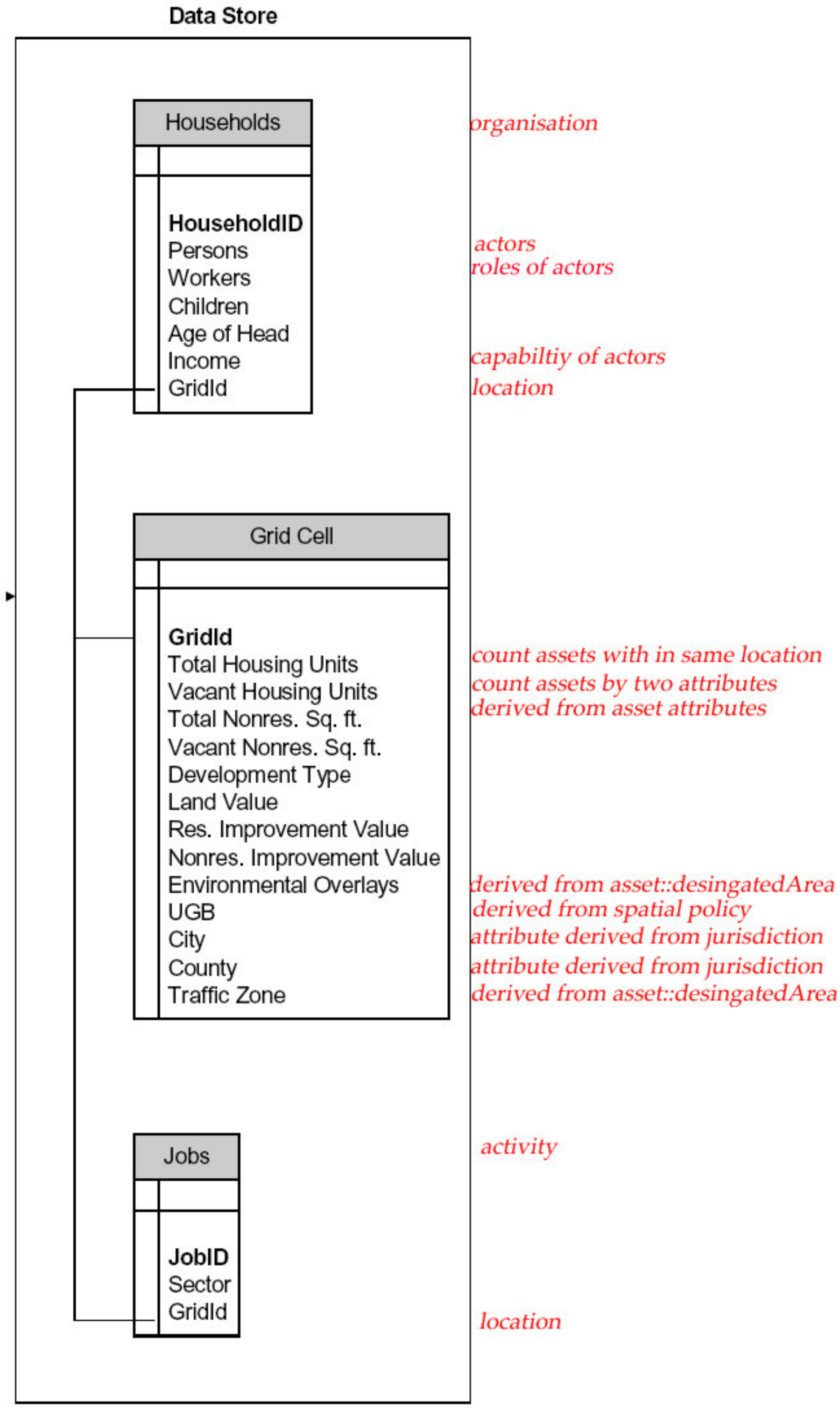

Figure 7 UrbanSim Data Store from (Waddell 2000) and representations in planning data model

Endogenising a policy that THE UGB is to be updated every five years to include a 20 year supply of developable land is also fairly straightforward using this structure. The following pseudo code illustrates this case:

UGB UPDATE POLICY

Antecedent Clause

If UGB (**designated Area *GML data***) exists (**logical operator**) for a city (**actor**)

and (**logical operator**)

if current (**timestamp**) UGB is 5 years old (**difference in timestamps**)

/*Existential operator has to check for the a previous instance of the UGB*/

Consequent clause

Metro Regional government (**actor**) will forecast for a period of 20 years (**Action based on the model**) demand (**indicator**)for different landuses(**activity**) and allocate (**action to be taken by the model**)the demand within current (**time stamp**)UGB

Else (**Antecedent Clause**)
If current (**time stamp**) UGB doesn't accommodate demand

Consequent Clause

Extend UGB (action) based on Plan information (pointer to Plan) and update (action) UGB regulation (Pointer to UGB regulation)

Redefinition of the UGB will be a task of updating the GML data based on forecasts and allocations of demand. Various sub-model forecasts and indicators, such as demand, must be recomputed at the time step, presumably by updating the data on which the forecast is based. That is, the forecast itself if also endogenous. Allocation of such demand within the UGB should be handled by different model to check if the current UGB will accommodate the forecasted 20 year demand. If not, plan information has to be available for a model to allocate appropriately such land use and thus change the UGB appropriately by incorporating additional land. Doing so will update the information that the UGB regulation and UGB update policy will apply in future time steps. As such a policy about a policy or regulation can be endogenized and will help in extending the predictions of different models.

Conclusion

Much work remains to achieve a shared data model that will enable representation of plans and regulations in a variety of simulation models of urban development. The current version is sufficient to begin trials to encode plans and regulations as fixed inputs, for interactive use, and for endogenous modelling. Tests of multiple modelling applications can begin by using the conceptual data model to translate inputs and outputs for partially substitutable models. The Planning Data Model clearly must be modified and extended based on the experience of such tests.

References

Deal, B. & George, R. V. (2001). Ecological Sustainability and Urban Dynamics: A Disaggregated Modeling Approach to Sustainable Design. 7th International Conference on Computers in Urban Planning and Urban Management., Honolulu, HI.

Forrester, J. W. (1961). *Industrial Dynamics*. Cambridge, MA, MIT Press.

Hannon, B. (1994).

Hopkins, L. D. (2001). *Urban development: The logic of making plans*. Washington, DC, Island Press.

Hopkins, L. D. (2003). Integrating knowledge about land use and the environment through use of multiple models. In S. Guhathakurta (Ed.), *Integrated urban and environmental models: A survey of current research and applications*. New York, NY, Springer-Verlag**:** 45-52.

Hopkins, L. D., Kaza, N. & Pallathucheril, V. G. (2003). Planning Markup Language: Representing the Meanings of Plans and Regulations. Association of Collegiate Schools of Planning/Association of European Schools of Planning, Leuven, Belgium.

Knaap, G. J. & Hopkins, L. D. (2001). "The Inventory Approach to Growth Management." *Journal of the American Planning Association* 67(3), 314-326.

Lake, R. (2000) Introduction to GML: Geography Markup Language. January, 2003, http://www.w3.org/Mobile/posdep/GMLIntroduction.html

Noth, M., Borning, A. & Waddell, P. (2003). "An Extensible, Modular Architecture for Simulating Urban Development, Transportation, and Environmental Impacts." *Computers, Environment and Urban Systems* 27(2), 181-203.

United States Environmental Protection Agency (2000). Projecting Land-Use Change: A summary of Models for Assessing the Effects of Community Growth and Change on Land Use Patterns. Cincinnati, OH, United States Environmental Protection Agency, Office of Research and Development.

Waddell, P. (2000). "A Behavioral Simulation Model for Metropolitan Policy Analysis and Planning: Residential Location and Housing Market Components of UrbanSim." *Environment and Planning B: Planning and Design* 27(2), 247-263.

Waddell, P. (2002). "UrbanSim: Modeling Urban Development for Land Use, Transportation and Environmental Planning." *Journal of theAmerican Planning Association* 68(3), 297-314.

Waddell, P. & Ulfarsson, G. F. (forthcoming). Introduction to Urban Simulation: Design and Development of Operational Models. , eds. Pergamon Press. In Stopher, Button, Kingsley

&Hensher (Ed.), *Handbook in Transport, Volume 5: Transport Geography and Spatial Systems*, Pergamon Press.
Westervelt, J. (2001). *Simulation Modeling for Watershed Management*. New York, NY, Springer-Verlag.

Appendix A: Data Model Component Diagrams.

The following diagrams elaborate parts of the data model by focusing on a particular entity type (object class) and its major relationships to other entities (object classes).

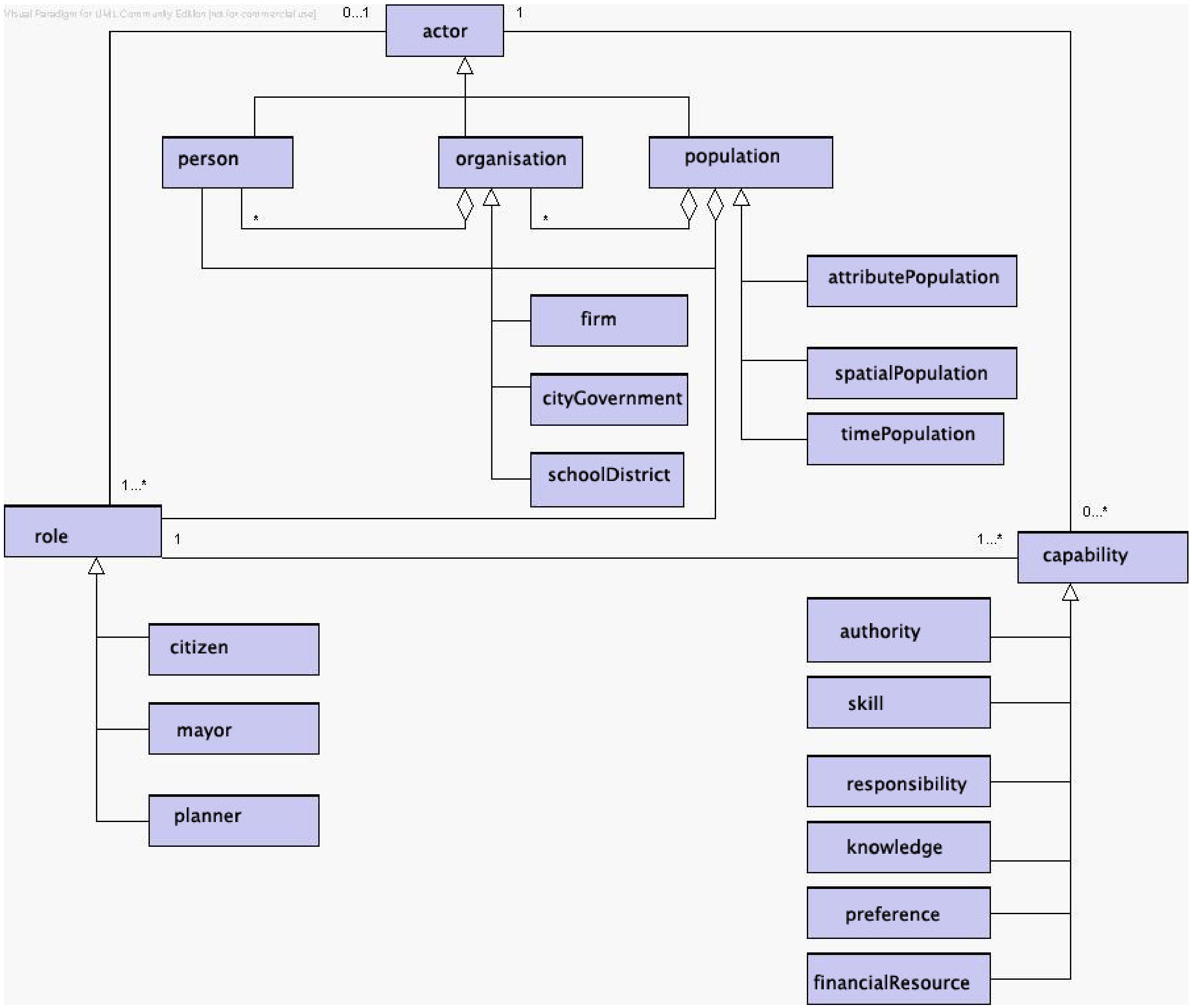

Figure A.1 Actors

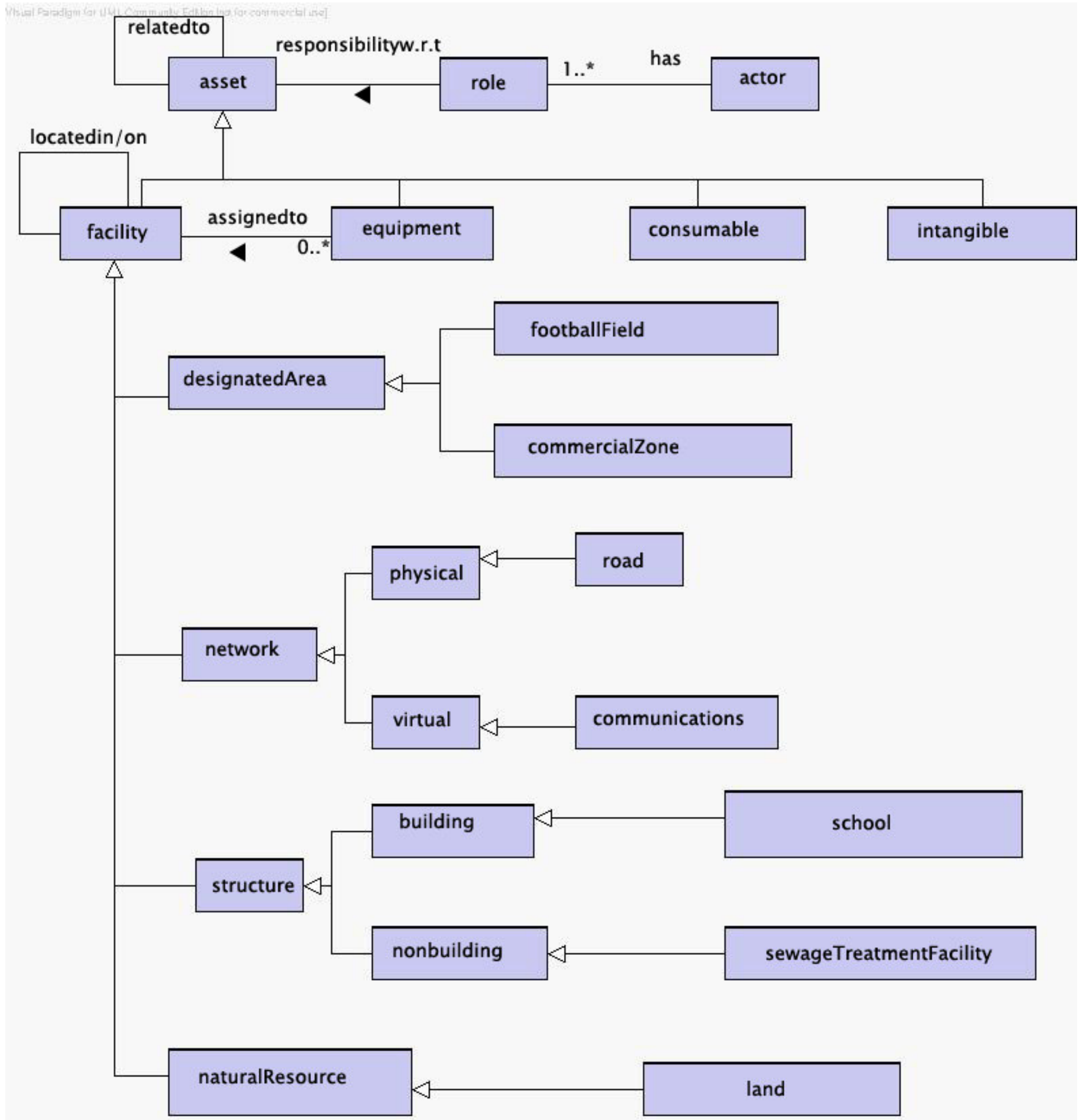

Figure A.2 Assets

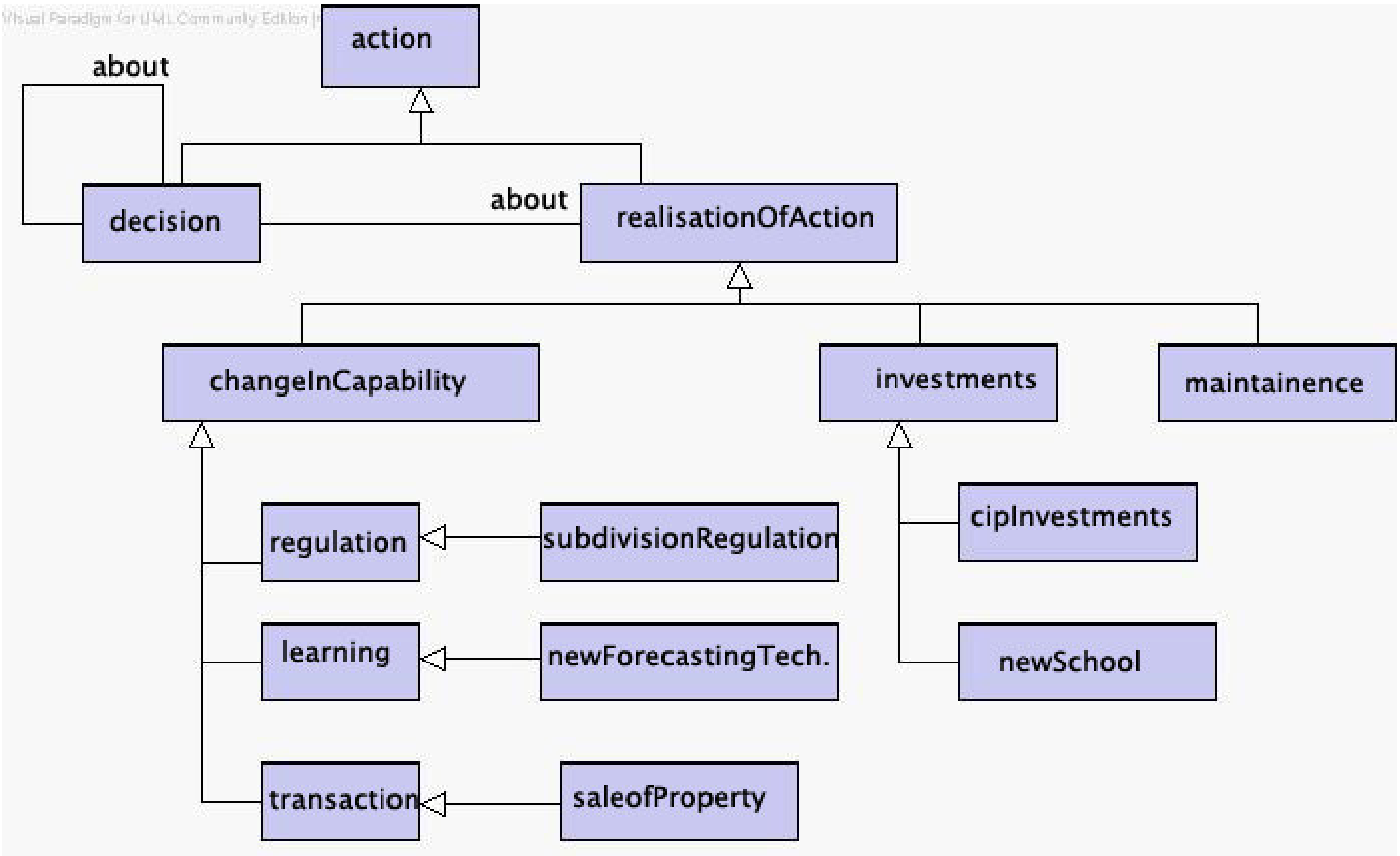

Figure A.3 Actions

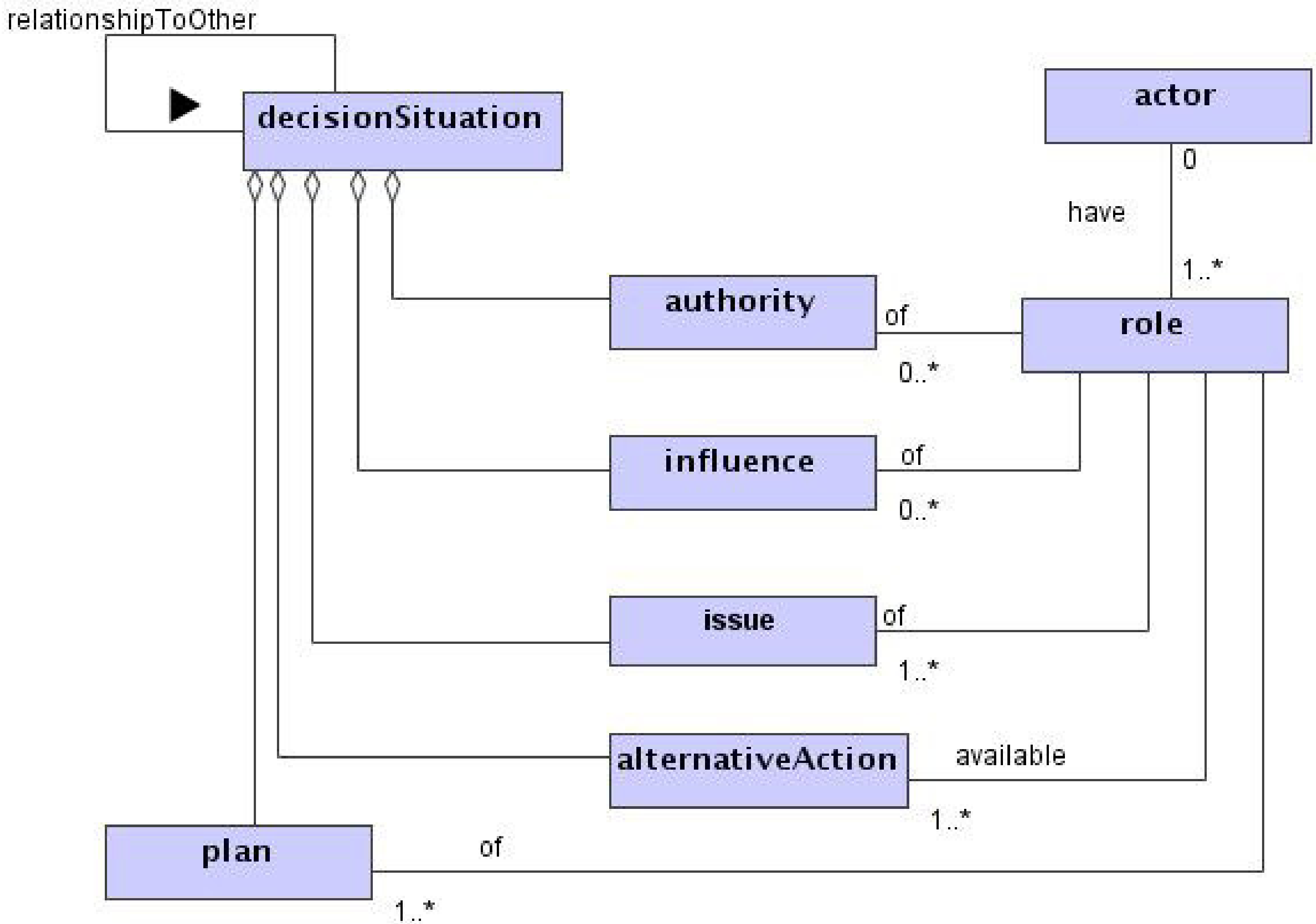

Figure A.4 Decision Situations

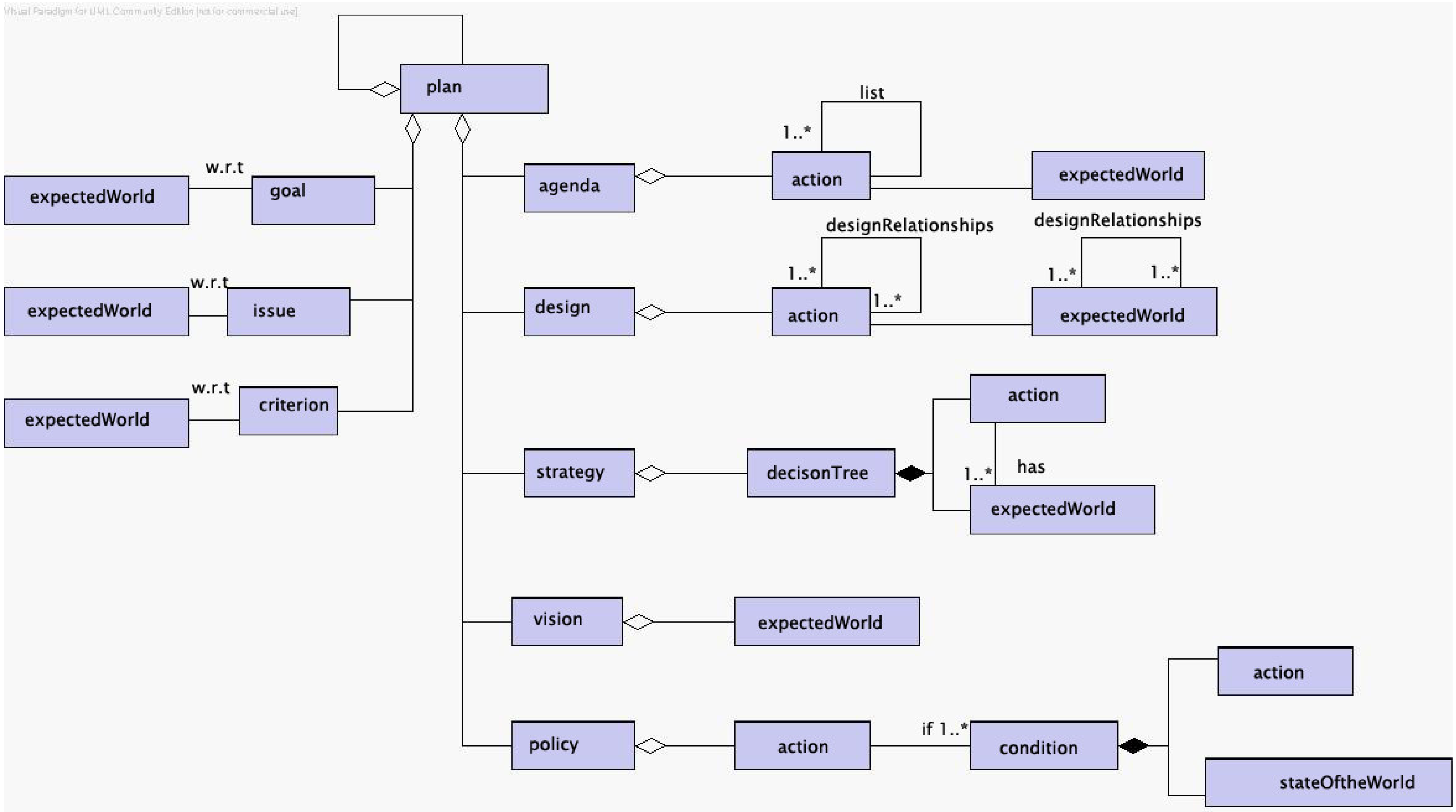

Figure A.5 Aspects of Plans